Architectural Approach for Implementing Access Control for Enterprise and Application Data Assets

Imran Quadri Syed

Lead Systems Developer, Information Technology
Prime Therapeutics, Eagan, Minnesota, USA

Abstract *— The purpose of this document is to explain in detail, architectural steps involved to implement access control for Enterprise and Application Assets on edge/gateway node for Hadoop (Big Data) cluster or on any ETL Server (DataStage/SSIS).*

Keywords *—EnterpriseAssets, Application Assets, Access Control, Security, Architecture, big data, Hadoop, ETL.*

I. INTRODUCTION

In today's world we are processing/storing huge volume of data using big data technologies like Hadoop, NiFi, HBase, Kudu, Hive and Spark. Processing large volume of data to meet business needs is critical at the same time it is important to properly secure this data. In this article we will walkthrough on how to secure the enterprise and application data assets on edge node or ETL servers (DataStage/SSIS).

Organizations are using more and more of publisher-subscriber (pub-sub) pattern. As part of pub-sub pattern the publisher publishes data files at predefined frequency for downstream applications to consume. The applications using these data are called subscribers. The major advantage of pub-sub pattern is, if there are 100 downstream applications that need data from a specific enterprise data source then they all can subscribe to published files, not using pub sub pattern would have resulted in 100 downstream applications connecting and querying the source system which would have created a tremendous load on the source system. While there are the advantages of pub sub pattern at the same time it is critical to ensure that the files that are being published are secured and only the approved subscribers have read only access. Subscribers should not be able to modify the published data. Apart from pub sub pattern, there are many other data files that reside on the servers and required to be properly secured. It is very imperative that this data is secured from unauthorized users and to make sure the service accounts have write access to only their own data domains.

Organizations assets at high level could be divided into 2 categories

- Enterprise Assets
- Application Assets

A. Enterprise Assets: Enterprise data assets are core data assets related to an organization that are critical for company's operations.For an example, health insurance company's core data domain would be eligibility and claims.Similarly, retail company's core data domains would be products, customer, stores and invoices.

B. Application Assets: Application assets are assets related to a specific application.For an example, a reporting application that would provide details of well performing stores by each state. For this application would need access to enterprise assets/data domains like stores and invoices.

To better understanddifference between enterprise and application assets, I would like to give an analogy with our day to day usage of smart phone. We have "contacts" in phone, it has phone number and other details of our contacts. Now we have messaging applications like WhatsApp, Viber,these apps use contacts to send text messages, make calls and group chats. The basic need for these applications is "contacts" information. So, in one sense you could think of contacts as critical data on phone for other applications to work (similar to enterprise assets for an organization). Applications like WhatsApp, Viber would access the contacts, but WhatsApp can't write into Viber's chat and vice versa.

In corporate world similarly, the applications would require read access to enterprise assets, but they should not have write permission to enterprise assets. Applications should be able to write to their own application assets/directories but not to another application directories/assets or enterprise directories/assets.

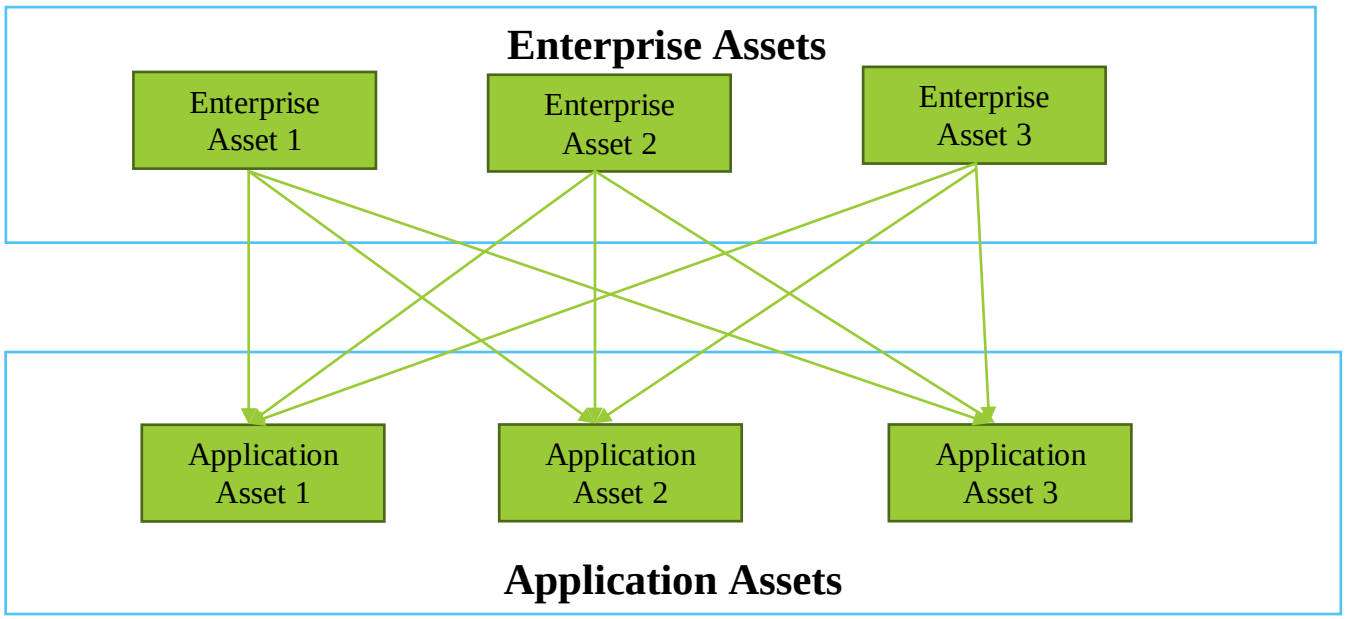

Figure 1: Enterprise Assets-Application Assets Relationships.

In the above example, application assets are accessing data from various enterprise assets, but one app is not able to access data of a different app. The applications can write data into their own application directories but can't write to any other application's directories.

II. ENTERPRISE ASSETS ACCESS CONTROL

Enterprise data assets are core data domains related to an organization. The Enterprise assets holds the critical information that is required by many applications. These enterprise assets should be accessible by other applications.

A. Mount point Level Directories:

Location	/edh	/edh/entrpsdata
Owner	Administrator Service Account	Administrator Service Account
Group	Administrator Group	Administrator Group
User Permission	rwx	rwx
Group Permission	rwx	rwx
Other Permission	r--	r--

The Unix mount point level directory "/edh" would be owned by Datahub/Datawarehouse administrator account because datahub admin would work with Unix team for applying any system patches and to troubleshoot any mount point issues. The next level directory "/edh/entrpsdata" is also owned by Datahub administrator account as well, as there are no specific enterprise data domains at this level to be owned by an enterprise data domain service account.

B. Enterprise Data Domain Level Directories:

Location	/edh/entrpsdata/ domain	/edh/entrpsdata/domain/ subdirectories
Owner	Enterprise Asset Service account	Enterprise Asset Service account
Group	Secure Group	Secure Group
User Permission	rwx	rwx
Group Permission	r-x	r-x
Other Permission	---	---

"/edh/entrpsdata/domain" is the enterprise data domain directory. At this level there could be multiple data domain directories like

1. /edh/entrpsdata/domain1
2. /edh/entrpsdata/domain2
3. /edh/entrpsdata/domain3

"/edh/entrpsdata/domain" and all sub directories "/edh/entrpsdata/domain/subdirectories" are owned by their corresponding enterprise service accounts and owner group for these directories would be "Secure Group".

Read access to enterprise assets could be granted by making the application service accounts part of secure group.

Example:

For better understanding, let's consider Retail Industry. In retail industry the core enterprise assets would be as follows

1. Products
2. Customers
3. Stores.
4. Invoice

Below are high level directories independent of enterprise data domains.

Mount Point Level Directories:

Location	/edh	/edh/entrpsdata
Owner	edhadmp	edhadmp
Group	edhadmp	edhadmp
User Permission	rwx	rwx
Group Permission	rwx	rwx
Other Permission	r--	r--

As discussed above the mountpoint directory "/edh" and "/edh/entrpsdata" would be owned by datahub administrator account. In this case "edhadmp" is the administrator account.

Understanding naming convention of service account "edhadmp" as follows:
"edh"represents"enterprise datahub". Use "edw" in case of "enterprise data warehouse".
"adm" refers to "administrator".
The last character represents the environment, in this case "p" represents production whereas for lower environments "s" would represent sandbox, "t" representing test/qa environment and "u" representing UAT environment.

From permissions perspective, the edhadmp id/group has read, write and execute permission at /edh/entrpsdata directory location. Others have read and execute permission.

Underneath mount level directories are the directories for "Products" and "Customers" enterprise data domains.

Enterprise Asset Directories (Products):

Location	/edh/entrpsdata/ products	/edh/entrpsdata/products/ extracts
Owner	edhpdcsp	edhpdcsp
Group	edhsecgp	edhsecgp
User Permission	rwx	rwx
Group Permission	r-x	r-x
Other Permission	---	---

Enterprise Asset Directories (Customers):

Location	/edh/entrpsdata/ customers	/edh/entrpsdata/products/ extracts
Owner	edhcstsp	edhcstsp
Group	edhsecgp	edhsecgp
User Permission	rwx	rwx
Group Permission	r-x	r-x
Other Permission	---	---

If you notice in the above table, the owner group for both the Product and Customer enterprise asset is set to **same group edhsecgp**. The edhsecgp group would be used to grant permissions to various application service accounts to read data from enterprise assets. **All the application service accounts would be part of edhsecgp** and if you notice edhsecgp has read and execute access but no write access. This would enable applications to read from enterprise asset directories but not be able to write to them. Only Enterprise assets service accounts (edhpdcsp&edhcstsp) could write to their corresponding enterprise asset directory.

Understanding naming convention of service accounts "edhpdcsp" and "edhcstsp" as follows:

"edh" represents "enterprise data hub"
"pdc" represents "Product" domain. "cst" represent "Customer" domain.
"s" represents "service account".
"p" represents "production" environment

Understanding naming convention of secure group "edhsecgp" as follows:

"edh" represents "enterprise data hub"
"sec" represents "secure".
"g" represents "unix group".
"p" represents "production" environment

As the products and customers are 2 separate data domains. The extracts from these 2 domains are

published in 2 different directory locations "/edh/entrpsdata/products/extracts" and "/edh/entrpsdata/customer/extracts". Each enterprise domain would have its own service account that would be able to publish/write data at these locations. In this case ofedhpdcsp for Product domain and edhcstsp for Customer domain.

III. APPLICATION ASSETS ACCESS CONTROL

Application assets are assets that are related to a specific application and only that specific application's service account would have access to write to this area.

A. Mount Point Level Directories:

Location	/edh	/edh/appdata
Owner	Administrator Service Account	Administrator Service Account
Group	Administrator Group	Administrator Group
User Permission	rwx	rwx
Group Permission	rwx	rwx
Other Permission	r--	r--

The highest Unix mount point level directory "/edh" is owned by Datahub/Datawarehouse administrator account because datahub admin would work with infrastructure team during application of system patches and to troubleshoot any mount point issues. The next level directory "/edh/appdata" is also owned by Datahub administrator account as well because "/edh/appdata" directory is not related to a specific application.

B. Application Level Directories:

Location	/edh/appsdata/ app	/edh/appsdata/app/ subdirectories
Owner	Service account	Service account
Group	Service account group	Service account group
User Permission	rwx	rwx
Group Permission	r-x	r-x
Other Permission	---	---

Underneath mount level directories are the application level directories. "/edh/appsdata/app" is the application directory. At this level there could be multiple application directories as follows

1. /edh/appsdata/app1
2. /edh/appsdata/app2
3. /edh/appsdata/app3

"/edh/appsdata/app" and all sub directories "/edh/appsdata/domain/subdirectories "are owned by its corresponding application service account and group owner for these directories would be application service account primary group. Application service account would have read, write and execute access on these directories.

Example:
Let's say we have 2 applications named "Sales Reporter" and "Core Analytics". These applications would require access to various enterprise data domains like Customers, products, stores, invoice to produce reports.

Mount Point Level Directories:

Location	/edh	/edh/appdata
Owner	edhadmp	edhadmp
Group	edhadmp	edhadmp
User Permission	rwx	rwx
Group Permission	rwx	rwx
Other Permission	r-x	r-x

As discussed in above sections, high-level directories "/edh" & "/edh/appdata" would be owned by edhadmp admin account.

Application Level Directories (Sales Reporter):

Location	/edh/appdata/ salesrerporter	/edh/appdata/salesreporter/ (archive, rawfiles, ctlfiles)
Owner	edhslrsp	edhslrsp
Group	edhslrgp	edhslrgp
User Permission	rwx	rwx
Group Permission	r-s	r-s
Other Permission	r-x	r-x

Underneath"/edh/appsdata/" directories, all applications would have their corresponding application directories. In this case "Sales Reporter" application have its own directories under "/edh/appsdata/salesreporter" that are owned by its own service account "edhslrsp". Similarly, "Core Analytics" application would have its own application directories under "/edh/appsdata/coreanalytics" that are owned by its corresponding service account "edhcrasp". Both the "edhslrsp" and "edhcrasp" would be part of "edhsecgp" that way they can have read only access to enterprise assets.

Application Level Directories (Core Analytics):

Location	/edh/appdata/ coreanlaytics	/edh/appdata/coreanalytics/ (archive, rawfiles, ctlfiles)
Owner	edhcrasp	edhcrasp
Group	edhcragp	edhcragp
User Permission	rwx	rwx
Group Permission	r-s	r-s
Other Permission	r-x	r-x

If you notice, all the directories of core analytics are owned by service account edhcrasp. "edhcrasp" would have read access to enterprise assets as it is part of edhsecgp group and it would have read, write, execute access to its own application directories related to core analytics.

Overall the approach here is, we are letting enterprise assets be available to all applications by making applications service accountsto be part of "edhsecgp" group at the same time we are making sure the applications could not modify the data in enterprise asset directory as unix group "edhsecgp" has only read access, also applicationsare able to only write to their own application directories and would not be able to read or modify data in other application directories.

IV. CONCLUSIONS

By implementing approach outlined in above article, organizations could ensure that the right security access is in place to meet security requirements and compliance.

V. REFERENCES

[1] Imran Quadri Syed "*Architectural Pattern for Implementing Data Quality Monitoring and Reporting Framework*" International Journal of Engineering Trends and Technology 68.2(2020):46-53

[2] Imran Quadri Syed "*Big Data Architectural Pattern to Ingest Multiple Sources and Standardization to Immune Downstream Applications*" International Journal of Engineering Trends and Technology 68.1(2019):5-10

[3] Fatimetou Zahra Mohamed Mahmoud, Noor Aziza Mohamadali*"The Business Intelligence Use In Healthcare And Its Enhancement By Predictive Analytics"* International Journal of Engineering Trends and Technology 67.7 (2019): 26-39

[4] Shrawan Kumar Sharma, Vijay Kumar Chhipa "*Security Issues, Challenging and Integration of IoT and Cloud Computing Technology*" International Journal of Computer Trends and Technology 68.4 (2020):48-53.

[5] Atif Mohammad, Hamid Mcheick, Emanuel Grant "*Big Data Architecture Evolution: 2014 and Beyond*" published in Association for Computing Machinery in September 2014

[6] Alrawda Abdullatif Abdulhaleem Hamid "*A Functional View of Big Data Ecosystem*" International Journal of Computer Trends and Technology 68.4 (2020):233-237